I claim:

1. A method for determining a drop probability, the method comprising:
systematically calculating a weight for determining a weighted moving average of a queue in a node;
calculating the weighted moving average;
determining an average queue size based upon the weighted moving average; and
evaluating a control function using the average queue size to determine the drop probability.

Sub A4

2. The method according to claim 1, wherein systematically calculating a weight comprises:
determining a sampling period for measuring the queue size;
determining a time period for which samples significantly contribute to the average queue size; and
determining the weight based upon the sampling period and the time period.

3. The method of claim 1, wherein determining a control function comprises:
determining a queue function based upon predetermined system parameters; and
determining the control function based upon the queue function.

4. The method according to claim 3 wherein determining the control function further comprises:
selecting a queue policy;
determining a threshold value based upon the selected queue policy
determining a maximum point based upon the threshold value, wherein the maximum point is outside of the queue function
selecting the control function such that when the control function is evaluated a point passes through the maximum point.

5. The method according to claim 4 wherein the queue policy is a delay conservative
policy and wherein determining a threshold value comprises:
determining a maximum value for the average queue size.

6. The method according to claim 4 wherein the queue policy is a drop conservative policy and wherein determining a threshold value comprises:

determining a maximum value for the drop probability.

7. A method for reducing oscillations in queue size in a link using congestion control that operates in a TCP environment, the method comprising:

determining a queue law function defining the average size for a link based at least upon the variable of drop probability;

defining a control function which determines the drop probability based upon the average queue size wherein a bounding point for the control function defining a maximum value of drop probability and a maximum value of the average queue size is greater than an equivalent point on the queue law function for either the maximum value of the average queue size or the maximum value of the average drop probability; and

dropping packets from the queue based upon a packet drop rate defined at a point of intersection for the control function and the queue law function.

8. The method according to claim 7, wherein in the step of defining the control function, the control function is further defined as a function having no discontinuities.

9. The method according to claim 7, wherein the function is piecewise linear.

10. A method for increasing utilization of a link capable of receiving a number of flows into a buffer, the link residing in a TCP network, the link having a congestion control module which drops packets to avoid buffer overflow, the method comprising:

determining a quantity representative of a capacity for the link;

calculating a quantity representative of the throughput for the link;

determining the utilization based on the capacity of the link, the throughput of the link, the numbers of flows and a packet drop probability; and

adjusting the packet drop probability to increase the utilization of the link.

11. A method for congestion control in server having a queue which resides in a

network wherein each data transmission from a sender to a receiver is sent at a transmission rate and the data transmission is acknowledged by the receiver, wherein if the data transmission is not acknowledged by the sender reduces the transmission rate, the method comprising:

ascertaining a network function which represents an average queue size of the queue based upon a server drop rate;

determining a control function for the server which produces an average queue size based upon a given server drop rate;

calculating an equilibrium point based upon the intersection of the network function and control function; and

setting the drop rate of the server to the equilibrium point.

12. An apparatus for determining a drop probability, the apparatus comprising:

a buffer for receiving data into a node forming a queue;

a weight module for systematically calculating a weight for determining a weighted moving average of the queue in a node;

a queue estimator for calculating the weighted moving average based on the weight and the received data in the queue and determining an average queue size based upon the weighted moving average; and

a processor for evaluating a control function using the average queue size to determine the drop probability.

13. The apparatus according to claim 12, wherein the weight module:

determines the weight by first accessing a sampling period for measuring the queue size and a time period for which samples significantly contribute to the average queue size.

14. The apparatus according to claim 12, further comprising:

a configuration module for determining a queue function based upon predetermined system parameters and determining the control function based upon the queue function.

15. An apparatus for reducing oscillations in queue size in a link using congestion control that operates in a TCP environment, the method comprising:

a queue law module for determining a queue law function based on system parameters defining the average queue size for a link based at least upon the variable of drop probability;

a control function module defining a control function which determines the drop probability based upon the average queue size wherein a bounding point for the control function defining a maximum value of the drop probability and a maximum value of the average queue size is greater than an equivalent point on the queue law function for either the maximum value of the average queue size or the maximum value of the average drop probability; and

a processor for dropping packets from the queue based upon a packet drop rate defined at a point of intersection for the control function and the queue law function.

16. An apparatus for reducing oscillations in queue size in a link using congestion control that operates in a TCP environment, the apparatus comprising:

a configuration module for systematically determining control function configuration parameters based upon traffic characteristics;

a control function module receiving the control function configuration parameters which define a control function and receiving an estimated queue size, the estimated queue size used in conjunction with the defined control function to determine a drop probability; and

a processor for dropping packets from the queue based upon a packet drop rate.

17. A computer program product for determining a drop probability, wherein the computer program product has computer code on a computer readable medium, the computer code comprising:

computer code for systematically calculating a weight for determining a weighted moving average of a queue in a node;

computer code for calculating the weighted moving average;

computer code for determining an average queue size based upon the weighted moving average; and

computer code for evaluating a control function using the average queue size to determine the drop probability.

18. The computer program product according to claim 17, wherein the computer code for systematically calculating a weight comprises:
computer code for determining a sampling period for measuring the queue size;
computer code for determining a time period for which samples significantly contribute to the average queue size; and
computer code for determining the weight based upon the sampling period and the time period.

19. The computer program product according to claim 17, wherein the computer code for determining a control function comprises:
computer code for determining a queue function based upon predetermined system parameters; and
computer code for determining the control function based upon the queue function.

20. The computer program product according to claim 19 wherein the computer code for determining the control function further comprises:
computer code for selecting a queue policy;
computer code for determining a threshold value based upon the selected queue policy computer code for determining a maximum point based upon the threshold value, wherein the maximum point is outside of the queue function
computer code for selecting the control function such that when the control function is evaluated a point passes through the maximum point.

21. The computer program product according to claim so wherein the queue policy is a delay conservative policy and wherein the computer code for determining a threshold value comprises:
computer code for determining a maximum value for the average queue size

22. The computer program product according to claim 21 wherein the queue policy is a drop conservative policy and wherein the computer code for determining a threshold value comprises:
computer code for determining a maximum value for the drop probability.

23. A computer program product for reducing oscillations in queue size in a node using congestion control that operates in a TCP environment, wherein the computer program product has computer code on a computer readable medium, the computer code comprising:

computer code for determining a queue law function defining the average queue size for a link based at least upon the variable of drop probability;

computer code for defining a control function which determines the drop probability based upon the average queue size wherein a bounding point for the control function defining a maximum value of the drop probability and a maximum value of the average queue size is greater than an equivalent point on the queue law function for either the maximum value of the average queue size or the maximum value of the average drop probability; and

computer code for dropping packets from the queue based upon a packet drop rate defined at a point of intersection for the control function and the queue law function.

24. The computer program product according to claim 23, wherein in the computer code for defining the control function, the control function is further defined as a function having no discontinuities.

25. The computer program product according to claim 23, wherein the function is piecewise linear.

26. A computer program product for increasing utilization of a link capable of receiving a number of flows into a buffer, the link residing in a TCP network, the link having a congestion control module which drops packets to avoid buffer overflow, wherein the computer program product has computer code on a computer readable medium, the computer code comprising:

computer code for determining a quantity representative of a capacity for the link;

computer code for calculating a quantity representative of the throughput for the link;

computer code for determining the utilization based on the capacity of the link, the throughput the link, the number of flows and a packet drop probability; and computer code for adjusting the packet drop probability to increase the utilization of the link.

27. A computer product for congestion control in a server having a queue which resides in a network wherein each data transmission from a sender to a receiver is sent at a transmission rate and the data transmission is acknowledged by the receiver, wherein if the data transmission is not acknowledged the sender reduces the transmission rate, wherein the computer program product has computer code on a computer readable medium, the computer code comprising:

computer code for ascertaining a network function which represents an average queue size of the queue based upon a server drop rate;

computer code for determining a control function for the server which produces an average queue size based upon a given server drop rate;

computer code for calculating an equilibrium point based upon the intersection of the network function and control function; and

computer code for setting the drop rate of the server to the equilibrium point.